

Automatic Decomposition and Efficient Semi-Structured Meshing of Complex Solids

Jonathan E. Makem, Cecil G. Armstrong and Trevor T. Robinson

School of Mechanical and Aerospace Engineering, Queen's University,
Belfast BT9 5AH, N. Ireland j.makem@qub.ac.uk

Abstract. In this paper, a novel approach to automatically sub-divide a complex geometry and apply an efficient mesh is presented. Following the identification and removal of thin-sheet regions from an arbitrary solid using the thick/thin decomposition approach developed by Robinson *et al* [1], the technique here employs shape metrics generated using local sizing measures to identify long, slender regions within the thick body. A series of algorithms automatically partition the thick region into a non-manifold assembly of long, slender and complex subregions. A structured anisotropic mesh is applied to the long/slender bodies and the remaining complex bodies are filled with unstructured isotropic tetrahedra. The resulting semi-structured mesh possesses significantly fewer degrees of freedom than the equivalent unstructured mesh, validating the effectiveness of the approach.

Key Words. Automatic decomposition, geometric reasoning, efficient meshing, metric field, anisotropic meshing.

1 Introduction

A major advantage of applying unstructured tetrahedral meshes to complex geometries for structural problems is the robustness of the current automatic tet mesh generators. One disadvantage is their limited capability for generating appropriate anisotropic stretched meshes. For CFD problems, meshes with very large aspect ratios can be routinely generated and adaptively refined [2]. However in structural problems a mesh density of one or two elements through the thickness of a thin sheet or across the cross-section of a long/slender region is often sufficient. This means that local, mesh-based approaches to anisotropic refinement are difficult. Alternatively, by generating a “mixed” or “hybrid” mesh consisting of structured meshes on specific regions of the model, combined with unstructured meshes on more complex areas, it is possible to achieve a dramatic reduc-

tion in degrees of freedom and thereby improve the efficiency of the analysis process. Consequently, this requires a process to partition the model into sub-regions where the different meshing strategies may be applied. This may be achieved by identifying easily mappable regions and sweepable volumes such as thin-sheets or long, slender sections. Structured anisotropic pentahedral or hexahedral meshes may be applied to these regions, and the remaining complex volumes in the model may be filled with unstructured tetrahedral elements to produce an efficient, semi-structured, mesh of the model.

Geometric properties of solids such as the medial object have been used in the past to decompose complex geometries into thick and thin sub-regions [1]. A structured thin-sheet mesh could be applied to the thin-sheet regions and merged with isotropic unstructured elements in the adjoining thick regions. Even though a significant reduction in degrees of freedom was achieved using this approach, it was less than expected as the long, slender areas of the thick-region consumed a lot of nodes when meshed with tetrahedra. However, if these long, slender regions could be identified, instead of being tet meshed they could be filled with a structured, swept mesh comprising anisotropic elements, thereby reducing the mesh density even further.

The objective of this research is to develop an automatic approach that uses an *a priori* knowledge of shape properties to identify long, slender regions in complex solids for the application of an efficient semi-structured mesh. Local sizing measures such as edge length and curvature, face width and curvature and local 3D thickness are employed to generate metric tensor fields which identify meshable sub-regions within complex volumes. Intelligent routines interrogate the geometry and automatically partition the body, isolating the long/slender regions from residual complex regions. Appropriate meshing strategies are applied to the respective sub-regions and the metric fields are then used to grade the mesh along the length of each slender region ensuring a smooth transition with isotropic elements in residual complex areas. The effectiveness of the approach is demonstrated on a complex model with a substantial reduction in degrees of freedom achieved.

The remainder of this paper is organized as follows: Section 2 reviews related work on geometric reasoning for meshing applications; Section 3 explains how the metric fields are generated using local sizing measures; Section 4 describes the approach for identifying the long/slender regions and partitioning the body into a non-manifold assembly of meshable sub-regions; Section 5 details the results for a complex model and makes a comparison between the semi-structured mesh and the equivalent unstructured tetrahedral mesh; Section 6 presents conclusions and future work.

2 Previous Work

Thakur *et al* [3] gives a comprehensive summary of the state of the art in CAD model simplification techniques for meshing. Research of particular note to the work in this paper is the prototype thick-thin decomposition process developed by Robinson *et al* [1] provided a capability for locating thin-sheet regions in complex solid geometries. This functionality is now available in the commercial CAE tool CADfix, by Transcendata [4]. The approach uses the 3D medial object (MO) [5] of the solid to determine the local thickness followed by a 2D MO computation on the mid-surfaces. By comparing the diameter of the 2D MO (which is an approximation of the lateral dimensions of the object) to the thickness of the 3D MO, it is possible to determine if the region is thick or thin. If the diameter of the inscribed circle on the 2D MO is large in comparison to the thickness, the region will be a thin sheet. Luo *et al.* [6] addressed the very similar problem of finding thin sections for the generation of prismatic p-version finite element meshes, using an octree-based approach to identify medial surface points and local thin sections.

Although efficiently meshing the thin-sheet regions significantly reduces the degrees of freedom in the model, it was noticed that there was still a lot of degrees of freedom consumed when long/slender regions such as flanges were filled with an unstructured tetrahedral mesh.

Price *et al* [7, 8] proposed a logical approach to firstly partition a complex 3D object into meshable regions and secondly apply a structured mesh to these regions. The medial surface was used to sub-divide the solid into hex-meshable subregions known as primitives. After all the subregions have been formed, each primitive is meshed using a midpoint subdivision technique [9]. Mesh compatibility between the primitives can be controlled by an integer programming technique described by Tam *et al* [10]. However, the approach has its limitations as it relies on a robust 3D medial object computation. Moreover, a comprehensive treatment of all possible shape features has not been developed.

Tchon *et al* [11] used an *a priori* knowledge of model geometry to generate a Riemannian metric based on local curvature and thickness. An isotropically refined octree grid is used as a support medium to generate the shape metrics. The anisotropic sizing information provided by the metrics is then used to refine the mesh. Some impressive results have been achieved with this approach with a dramatic reduction in element count. However, the technique relies on an initial mesh and the quality of the elements generated after the refinement process is questionable as their shape is not perfectly cubical. Other authors [12, 13] have reported similar

geometry-based methods for mesh adaption and refinement. However, all of them relate to the relocation or refinement of an initial mesh. The focus here has been to use similar shape measures to identify regions of the model for structured meshing, rather than local mesh.

Zhao *et al* [14, 15] proposed a technique for the adaptive generation of an initial mesh based on the geometric features of a solid model. A special refinement field based on curvature and thickness was constructed to control mesh size and density distribution. The boundary of the adapted indentation mesh produced was matched to the boundary of the solid model using a threading method. Although the resultant mesh on a complex model possessed fewer degrees of freedom than a structured swept mesh applied to a manually sub-divided version, the quality of the mesh produced was not satisfactory as a significant number of singularities were evident.

White *et al* [16] developed an interesting algorithm for automatically decomposing multi-sweep volumes into volumes that can be swept meshed. This is achieved by discretizing the linking faces of volumes with quad elements which provide a layering system to traverse the volume vertically. Target faces are pushed through the volume onto opposing source faces providing an imprint that governs the decomposition of the solid. The newly generated volumes are all sweep meshable with a single target face. However, the major limitation with the approach is that it can only be applied to sweepable geometries and cannot be used to decompose more complex models. Consequently, this identifies a requirement for a more robust technique for the automatic decomposition of complex models that will successfully partition any dumb solid geometry into an assortment of sub-regions, including sweepable volumes, where various structured meshing strategies may be applied.

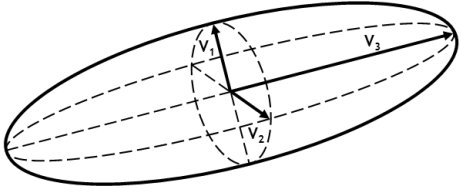
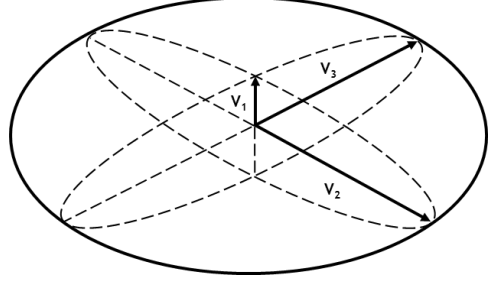
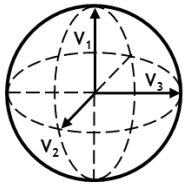
3 Geometric Reasoning using Local Sizing Measures.

The approaches and methodology reported in the forthcoming sections have been implemented in the CADfix software package using an API based on the Python programming language. Assuming that the thin-sheet regions have been identified and partitioned out using the aforementioned thick/thin decomposition tool developed by Robinson *et al* [1], decomposition of the remaining thick region into long/slender and chunky bodies may be achieved using a shape metric.

3.1 Metric Classification

The goal is to generate within the CAD model ellipsoids representing the shapes of different regions. The principal axes of an ellipsoid may be used to define the target element size in three dimensions [17].

Table 1. Ellipsoid classification

Ellipsoid Type	Diagram	Criteria
Long/ Slender		$V_3 \gg V_1$ $V_3 \gg V_2$ $V_1 \approx V_2$
Thin- sheet		$V_3 \gg V_1$ $V_2 \gg V_1$ $V_2 \approx V_3$
Isotropic		$V_1 \approx V_2 \approx V_3$

If an ellipsoid has one principal direction that is much greater than the other two, this identifies a slender region where the mesh can be extruded in the direction of the largest principal axis. Conversely, areas of the model where each axis is similar in magnitude and the ellipsoid is spherical identify regions where the target element size will be similar in all directions, requiring an unstructured isotropic tetrahedral mesh. The three main types

of ellipsoids are displayed in table 1. Note that regions represented by thin-sheet ellipsoids will have been removed by the thick/thin tool prior to implementing the approach presented here.

Within the procedures described in this paper, an ellipsoid is generated on every edge with its centre at the mid-point of the edge. The edge length and curvature are used to determine the length of the ellipsoid in the edge direction. In terms of long/slender ellipsoids, this sizing measure is normally employed to gauge the size of the largest principal axis which provides the sweep direction for the extruded mesh. The other sizing measures of face width and curvature and local 3D thickness are employed to determine the extent of the remaining two principal axes. If a face is planar, the length of an axis may be calculated using the 2D medial axis on the face or an approximation to it, whereas the axis length on non-planar surfaces may be determined using a curvature based searching algorithm for a pre-defined sag value, d . For other scenarios such as concave edges and corners ray casting is used to assess axis length by giving an approximation of the local thickness.

3.2 Metric Sizing using Edge Length and Curvature

If the edge is straight, then half the edge length is used for the extent of the ellipsoid axis by default, as shown in figure 1. However, if an edge is curved, it is necessary to employ a curvature sensitive approach to check the size of the ellipsoid axes along curved edges and surfaces. Along an edge, the change in tangent vector, \mathbf{T}_i , quantified by the angle α defined in equation 1, is used to determine the size of the first ellipsoid axis, \mathbf{V}_i , as shown in figure 2. When α reaches a tolerance value the ellipsoid axis vector \mathbf{V}_i can be determined.

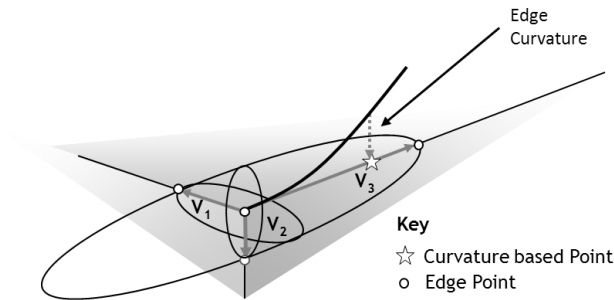


Fig. 1. 1D metric generated using edge length and curvature

$$\alpha = \cos^{-1} \left(\frac{T_i \cdot T_{i+1}}{|T_i| |T_{i+1}|} \right) \quad (1)$$

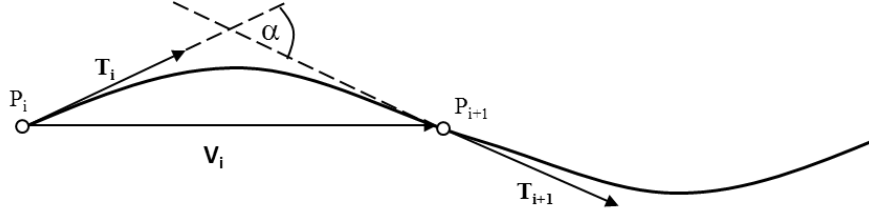


Fig. 2. Metric sizing using edge curvature

3.3 Metric Sizing using Face Width and Surface Curvature

If the surfaces adjoining an edge are curved, a similar strategy is applied to assess the size of the second and third axes. The surface curvature is used to determine the size of the ellipsoid axis based on a sag value, d . For a pre-defined d value, the size of the vector orthogonal to the tangent vector at the centre point on the edge can be calculated, as shown in figure 3. In cases where the surface is planar, the 2D medial object is used to determine the local face width and the length of the ellipsoid axis in this direction.

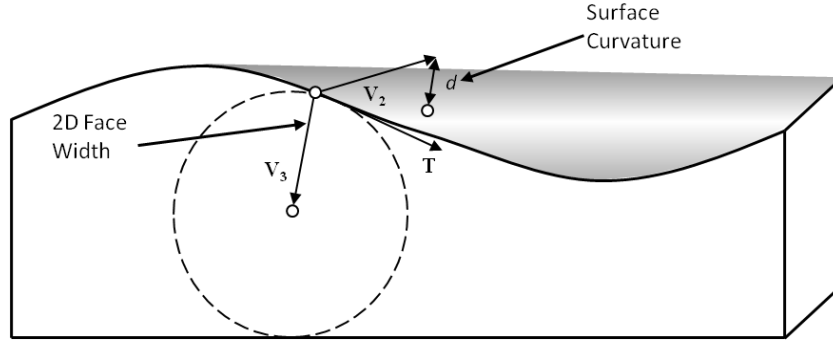


Fig. 3. Sizing of ellipsoid axis using surface curvature and face width.

On a parametric surface, as shown in figure 4, for a given tangent vector, T and normal vector N at point P on a given edge, the sag value, d at point Q on the surface is equivalent to the projection of PQ onto N , as described in equation 2.

$$d = \mathbf{PQ} \cdot \mathbf{N} \quad (2)$$

d can also be described as the perpendicular distance from Q to the tangent plane at P . In parametric terms, P and Q can be represented by the points $\mathbf{x}(u, v)$ and $\mathbf{x}(u + du, v + dv)$. Consequently, using Taylor's theorem [18], equation 3 can be modified to give:

$$d = \frac{1}{2} \mathbf{d}^2 \mathbf{x} \cdot \mathbf{N} + O(du^2 + dv^2) \quad (3)$$

Thus, for a pre-defined sag value, d , equation can be used to determine the location of point Q and the extent of the ellipsoid axis, \mathbf{V} .

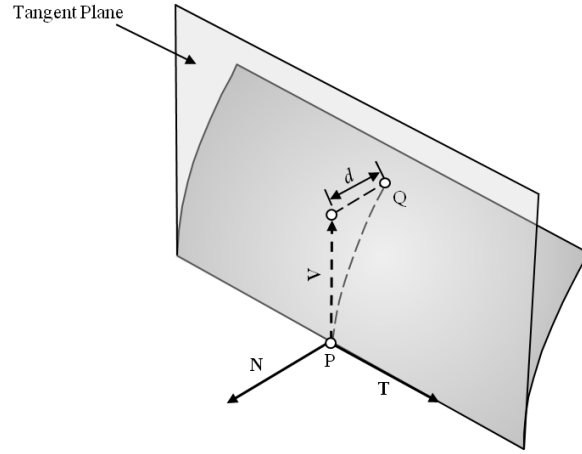


Fig. 4. Metric sizing for a given sag value on a parametric surface.

3.4 Metric Sizing using Local Thickness

Local 3D thickness is used as a sizing measure to gauge the extent of the ellipsoid axis for scenarios like concave and convex edges. For ellipsoids generated on concave or convex edges ray casting is used to assess the thickness of the body in that region, as shown in figure 5. An axis length based on the thickness of the body is then compared to potential axes generated using other sizing measures and the shortest axis is always selected for the ellipsoid in that direction.

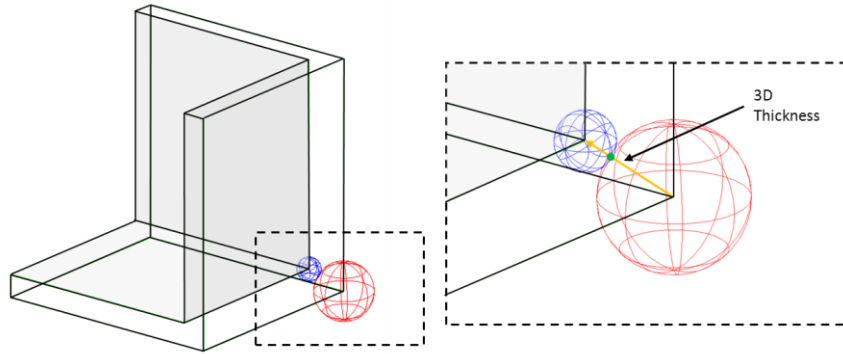


Fig. 5. Metric sizing by assessing local 3D thickness.

4 Automatic Partitioning of Complex Geometry

After a thick/thin decomposition has been performed on a complex geometry, as shown in figure 6, a search is then performed on the thick body (in red) for slender regions by generating ellipsoids at the mid-point of every edge.

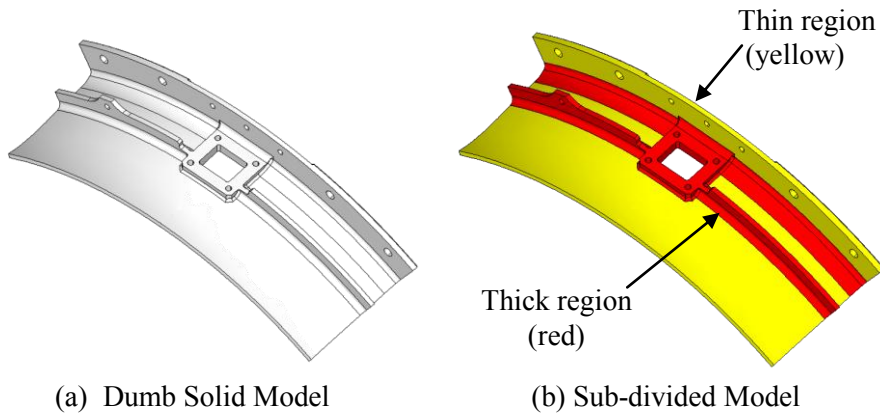


Fig. 6. Thick/thin decomposition of complex model

4.1 Identification of Long/Slender Regions

The aspect ratio of an ellipsoid compares the length of its longest axis to the length of its other axes. Critical aspect ratio (CAR) is a user specified

criteria that is used to determine above which ratio it is considered an ellipsoid axis is much longer than the other two. For a pre-defined critical aspect ratio (CAR), all long/slender ellipsoids are identified using the algorithm detailed in figure 7. Where D represents the dimension of the ellipsoid (for long/slender ellipsoids $D = 1$), V_1 , V_2 and V_3 are vectors defining the three principal axes of the ellipsoid, A is the vector with the greatest magnitude and B defines a set of vectors excluding the vector with the greatest magnitude. All of these long/slender ellipsoids for the thick region on the model shown in figure 6 are shown in figure 8.

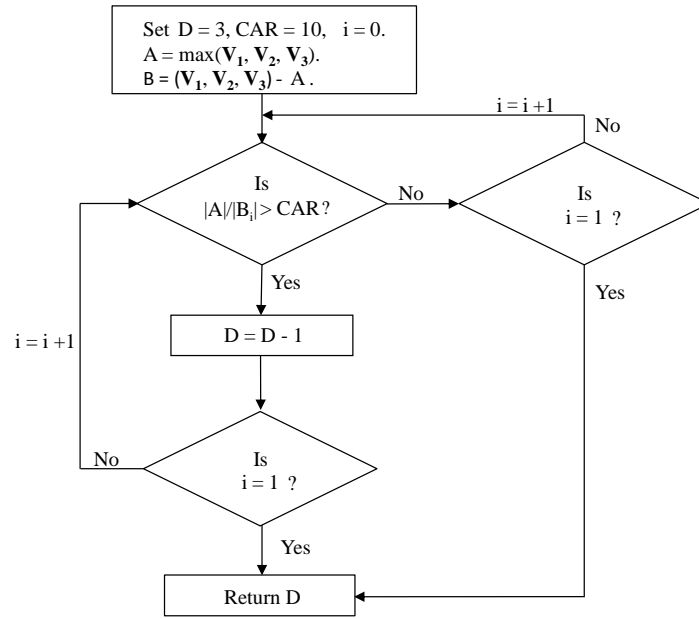


Fig. 7. Selection process for 1D ellipsoids using a critical aspect ratio.

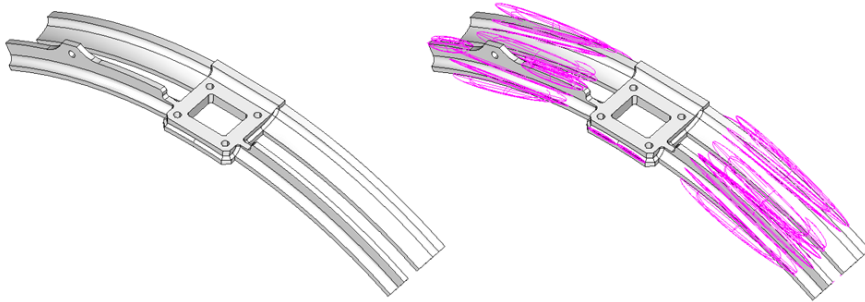


Fig. 8. Long/slender ellipsoids on the thick region of complex model.

After all the long/slender ellipsoids are generated, a “closed-loop” searching algorithm is then initiated that uses the ellipsoids to look for closed loops of surfaces which each bound a slender section. For any given edge on which a long/slender ellipsoid is located, the algorithm searches for an adjoining surface that has another long/slender ellipsoid on one of its bounding edges. If such an ellipsoid exists, this establishes a direction in which to continue the search, in either a clockwise or anti-clockwise fashion. The search is repeated for the other surface and if successful the process continues until a surface is located that contains an edge with the ellipsoid that was used for the initial iteration. At this point, a long/slender region comprising a closed loop of surfaces is identified and these surfaces are then removed from the search. A new edge is selected and the process is repeated until all closed loop surface groups have been identified in the model. Table 2 describes the closed-loop searching algorithm for long/slender extruded regions. The results of the closed-loop search are shown in figure 9. The algorithm successfully identifies five slender regions, where each region is highlighted in a different colour. Note that the bottom-left arm of the model is composed of two separate long/slender regions. The different surfaces identifying those regions are in different colours on the left and right image.

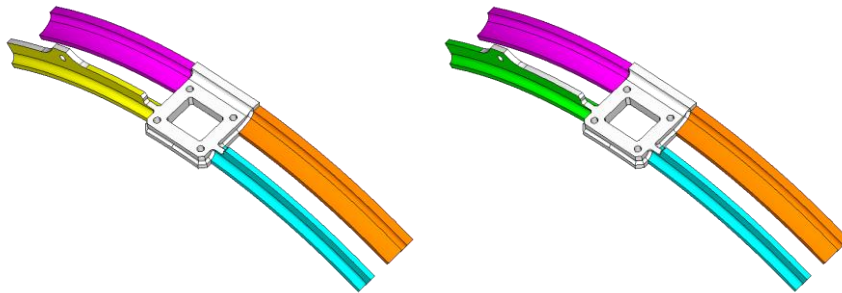
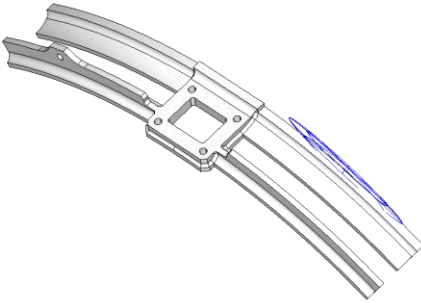
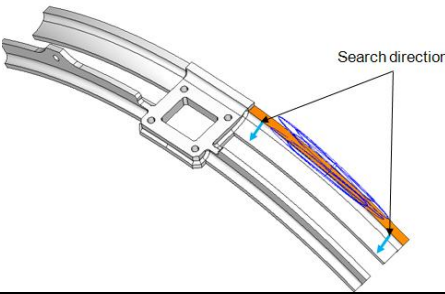
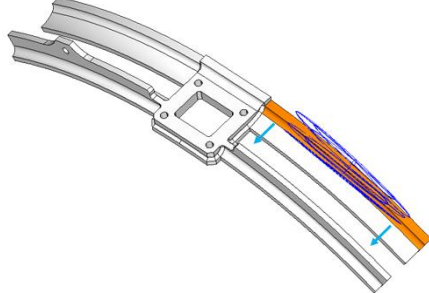
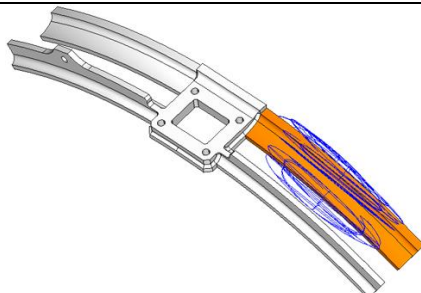


Fig. 9. Long/slender regions identified on complex model

Table 2. Closed-loop searching algorithm.

Stage	Diagram
1. Select an ellipsoid.	
2. Locate an ellipsoid on the edge of an adjoining surface.	
3. Continue the search in the search direction.	
4. Stop when a closed loop is formed	

4.2 Automatic Partitioning into Meshable Sub-Regions

After the groups of surfaces that define each slender region have been identified, the next stage of the process involves generating meshable slender bodies by partitioning the surface groups using cutting planes. Open ended slender regions, which don't require any cutting planes, can be identified because the 'capping' surface shares an edge with all of the surfaces bounding the sweepable region. Cutting planes are generated using the edge tangent and surface normal at an offset (calculated as a fraction of the shortest edge length) away from the long/slender-complex region interface, as shown in figure 10. The cutting plane(s) for each edge are grouped into a set and given an ID based on the surface group.

After associating each surface group with a cutting plane, the body is automatically partitioned into a non-manifold assembly of long/slender and residual complex bodies in CADfix, using series of cutting commands. The cutting plane ID's are then used to identify if a body is chunky or slender. Any body that lies between the cutting planes in a set is slender. The result of the automatic partitioning process is shown in figure 11.

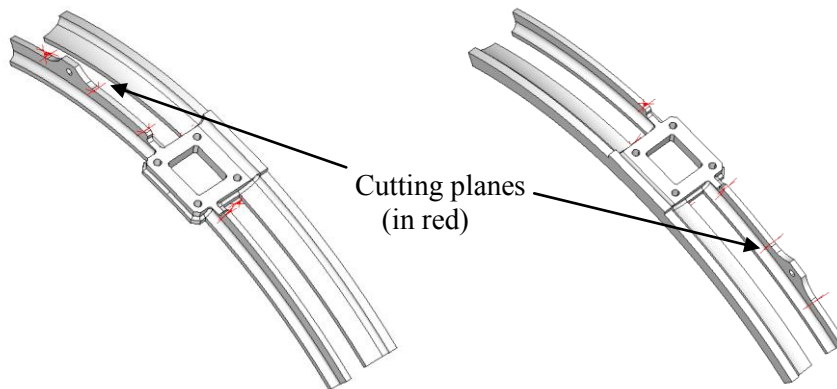


Fig. 10. Automatic generation of cutting planes

Finally, by combining the partitioned thick region with the thin region generated from the thick/thin decomposition it's possible to achieve the complete decomposition of the model into long/slender, thin-sheet and residual complex sub-regions, as shown in figure 12.

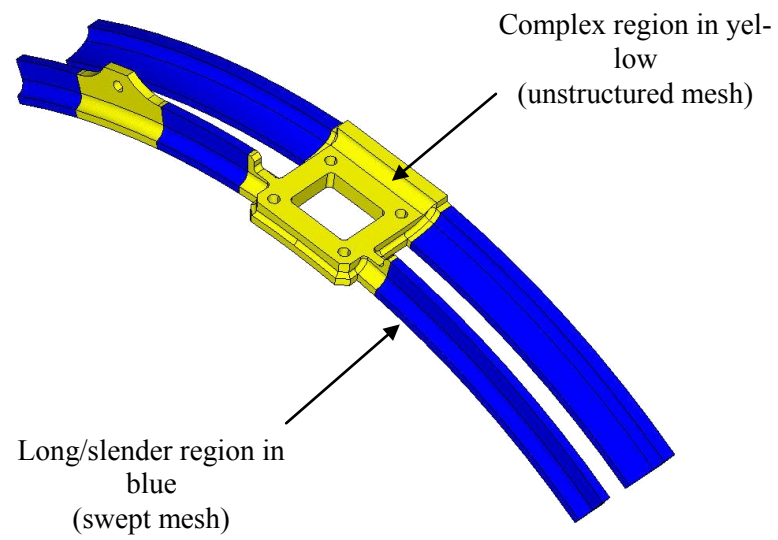


Fig. 11. Automatic partitioning into long/slender and complex sub-regions

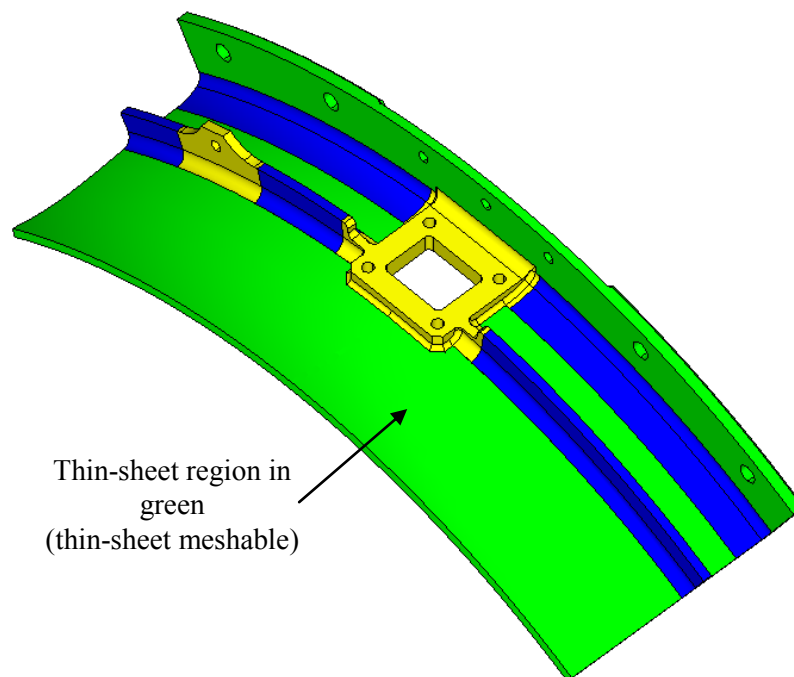


Fig. 12. Full decomposition of complex model

5 Meshing

After the thick region has been sub-divided a semi-structured mesh can be applied using various meshing algorithms. An extruded mesh is applied to the 1D regions using a 1D sweeping algorithm. The 3D complex volumes are filled with an unstructured isotropic tetrahedral mesh. Some refinements to ensure better matching of element size between 3D unstructured and 1D sweep meshes are planned. A 48% reduction in degrees of freedom is achieved with the semi-structured mesh (15,773 nodes) of the non thin-sheet region, shown in figure 13 compared to the unstructured tetrahedral mesh (30,108 nodes) of the component shown in figure 14.

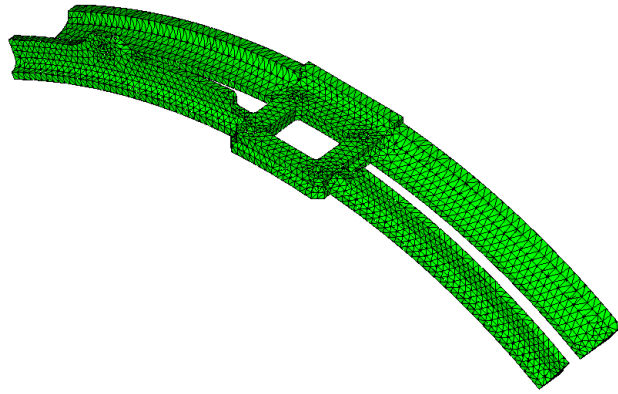


Fig. 13. Unstructured tetrahedral mesh of thick region

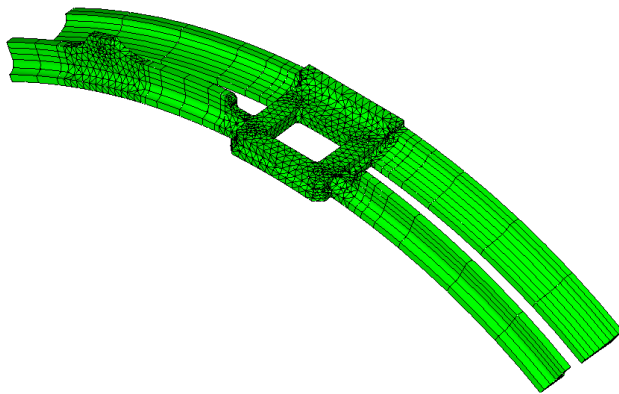


Fig. 14. Semi-structured hybrid mesh of thick region

The ellipsoids are not only used to partition the model but also grade the mesh in the long/slender regions. Each ellipsoid on the edge of a 1D sweepable region is used to define a “source field” on the model in CAD-fix. These source fields control the growth of the mesh and ensure a smooth transition between the 1D and 3D regions. In any case, the effectiveness of the mesh adaption technique is clearly demonstrated, generating an efficient mesh with a substantial reduction in degrees of freedom.

6 Conclusions

An automatic approach for sub-dividing a general dumb solid into meshable sub-regions has been developed. The approach relies on a robust thick/thin decomposition of the model. Provided the thick bodies can be identified, they can be partitioned into slender (1D) and chunky (3D) regions. Consequently, by applying a different meshing strategy to each region in the assembly of thin-sheet (2D), long/slender (1D) and chunky (3D) bodies, a semi-structured mesh can be generated, producing significantly fewer degrees of freedom and thereby enhancing the efficiency of the numerical analysis process.

Novel techniques for long/slender region identification have been employed that use shape metrics based on local sizing measures such as edge length, surface curvature, face width and local 3D thickness. After building an association between the shape metrics and the model geometry, a “closed loop” searching algorithm successfully locates closed loops or groups of surfaces that form slender regions within the model. Additional algorithms analyse these surface groups and generate cutting planes in strategic locations that are then used to automatically partition the model to produce a fully sub-divided, non-manifold assembly of meshable bodies. Future work will focus will be on testing the approach on assemblies of industrial complexity, quantifying the accuracy of the highly stretched meshing produced and the reduction in degrees of freedom attained.

8 Acknowledgements

The research leading to these results has received funding from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 234344 (www.cresendo-fp7.eu). The authors would like to acknowledge Transcendata for their support throughout the

course of the work and industrial partners for their support in providing complex models.

9 References

1. Robinson, T.T., Armstrong, C.G., Fairey R. Automated mixed dimensional modelling from 2D and 3D CAD models. *Finite elements in analysis and design*. 2011; (47) 151 - 165.
2. Loseille, A., Dervieux, A., Alauzet, F. Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations. *Journal of Computational Physics*. 2010; (229) 2866 - 2897.
3. Thakur, A., Banerjee, A.G., Gupta, S.K. A survey of CAD model simplification techniques for physics-based simulation applications. *Computer-Aided Design*. 2009; (41) 65 - 80.
4. CADfix, Transcendata. <http://www.transcendata.com/products/cadfix/>. (accessed 04/07/11)
5. Blum, H. A transformation for extracting new descriptors of shape. *Models for the Perception of Speech and Visual Form*. 1967. MIT Press, Cambridge, MA. Pp. 362 - 380.
6. Luo, X-J, Shephard, M.S., Yin, L-Z., O'Bara, R.M., Nastasi, R., Beall, M.W. Construction of near optimal meshes for 3D curved domains with thin sections and singularities for p-version method. *Engineering with Computers*. 2010; (26) 215-229.
7. Price M.A., Armstrong C.G. Hexahedral mesh generation by medial surface subdivision: part I. Solids with convex edges. *International journal for numerical methods in engineering*. 1995; (38) 3335 - 3359.
8. Price M.A., Armstrong C.G. Hexahedral mesh generation by medial surface subdivision: part II. Solids with flat and concave edges. *International journal for numerical methods in engineering*. 1997; (40) 111 - 136.
9. Li, T.S., McKeag, R.M., Armstrong, C.G. Hexahedral meshing using midpoint subdivision and integer programming. *Computer methods in applied mechanics and engineering*. 1995; (124) 171 - 193.
10. Tam T.H.K, Armstrong C.G. Finite element mesh control by integer programming. *International journal for numerical methods in engineering*. 1993; (36) 2581-2605.
11. Tchon, K., Khachan, M., Guibault, F., Camarero, R. Three-dimensional anisotropic geometric metrics based on local domain curvature and thickness. *Computer-Aided Design*. 2005; (37) 173 - 187.

12. Shimanda, K., Mori, N., Kondo, T. Automated mesh generation for sheet metal forming simulation. *Int J Vehicle Des.* 1999: (21) 278 - 291.
13. Frey, P.J. About surface meshing. Ninth International Meshing Roundtable, Sandia National Laboratories, New Mexico, pp 123 - 126.
14. Zhao, G., Zhang, H. Adaptive hexahedral mesh generation based on local domain curvature and thickness using a modified grid-based method. *Journal of Finite Elements in Analysis and Design.* 2007: (43) 691 - 704.
15. Zhao, G., Zhang, H., Cheng, L. Geometry-adaptive generation algorithm and boundary match method for initial hexahedral element mesh. *Journal of Engineering with Computers.* 2008: (24) 321 - 339.
16. White, D.R., Saigal, S., Owen, S.J. CCSweep: automatic decomposition of multi-sweep volumes. *Journal of Engineering with Computers.* 2004: (20) 222 - 236.
17. Frey, P.J., George, P.J. *Mesh Generation.* 2008. Wiley. pp 337.
18. Lipschutz, M.M. *Schaums Outline of Theory and Problems of Differential Geometry.* 1969. McGraw-Hill. pp 176.